# Ejercicio sobre JPA.

Requisitos: Usar buenas prácticas. Recordar que los controladores (controllers) NO deben tener ninguna referencia a la base de datos.

Tiempo máximo para hacerlo: El doble del que le haya llevado al primer estudiante (aprox: 6-8 horas)

Subirlo al repositorio de github, en la carpeta “examen\_JPA\_cascada”

Valoración: 10, 8, 6, 4, 3, 0.

1. Crear estas tablas en JPA con una base de datos h2, dejando el fichero en disco duro.

* CabeceraFra // Guardara las cabeceras de las facturas.

Id int autoincremental, -- ID

cli\_codi int - Codigo de cliente.

ImporteFra Double – Importe total de la fra.

* LineasFra // Las líneas de la fra.

Id int autoincremental, -- ID

IdFra int -- // Id de la fra.

ProNomb String not null , -- Nombre de producto.

Cantidad double,

precio double,

* Cliente

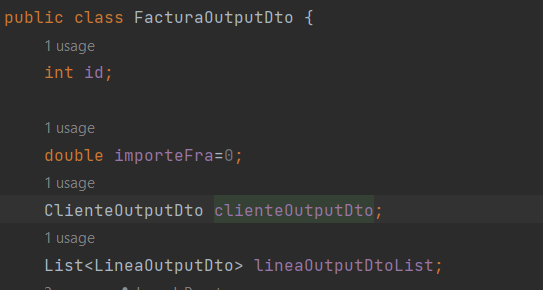
Id int autoincremental, -- ID

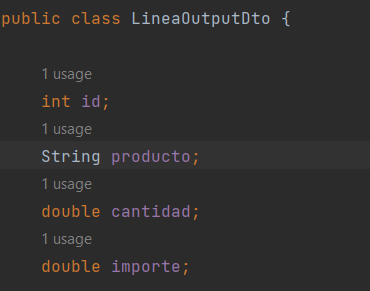
nombre varchar(100) not null -- // Nombre de cliente

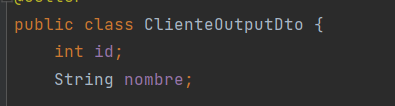
1. En cada ejecución:
   * Regenerar la base de datos (drop tables y create tables)
   * Insertar un nuevo cliente.
   * Insertar una factura de prueba asignada al cliente (anteriormente creado) con dos líneas. IMPORTANTE: Solo se debe realizar un **save** para insertar las 2 líneas y la cabecera de fra.

## Endpoints:

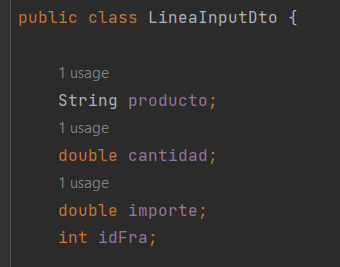
1. GET /factura. Buscar todas las facturas. Devolverá una List de FacturaOutputDto que debe contener estos campos. Siempre devolverá un 200.







1. DELETE factura /{idFra}. Borrar toda la fra. Deberá borrar con una sola instrucción tanto la cabecra como las lineas. Devolverá un 404 si no encuentra la fra. Un 200 en caso contrario.
2. POST factura/linea/{idFra}. Añadir una línea a una fra. ya existente. En el body recibira un objeto LineaInputDto



Devolverá el objeto FacturaOutputDto con la nueva línea añadida.

Si no existe la fra (id no existente) devolver un 404.